

SERVLETS - HITS COUNTER

<http://www.tutorialspoint.com/servlets/servlets-hits-counter.htm>

Copyright © tutorialspoint.com

Hit Counter for a Web Page:

Many times you would be interested in knowing total number of hits on a particular page of your website. It is very simple to count these hits using a servlet because the life cycle of a servlet is controlled by the container in which it runs.

Following are the steps to be taken to implement a simple page hit counter which is based on Servlet Life Cycle:

- Initialize a global variable in init() method.
- Increase global variable every time either doGet() or doPost() method is called.
- If required, you can use a database table to store the value of global variable in destroy() method. This value can be read inside init() method when servlet would be initialized next time. This step is optional.
- If you want to count only unique page hits with-in a session then you can use isNew() method to check if same page already have been hit with-in that session. This step is optional.
- You can display value of the global counter to show total number of hits on your web site. This step is also optional.

Here I'm assuming that the web container will not be restarted. If it is restarted or servlet destroyed, the hit counter will be reset.

Example:

This example shows how to implement a simple page hit counter:

```
import java.io.*;
import java.sql.Date;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class PageHitCounter extends HttpServlet{

    private int hitCount;

    public void init()
    {
        // Reset hit counter.
        hitCount = 0;
    }

    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
                      throws ServletException, IOException
    {
        // Set response content type
        response.setContentType("text/html");
        // This method executes whenever the servlet is hit
        // increment hitCount
        hitCount++;
        PrintWriter out = response.getWriter();
        String title = "Total Number of Hits";
        String docType =
            "<!doctype html public \"-//w3c//dtd html 4.0 \" +
            \"transitional//en\">\n";
```

```

        out.println(docType +
            "<html>\n" +
            "<head><title>" + title + "</title></head>\n" +
            "<body bgcolor=\"#f0f0f0\">\n" +
            "<h1 align=\"center\">" + title + "</h1>\n" +
            "<h2 align=\"center\">" + hitCount + "</h2>\n" +
            "</body></html>");
    }
    public void destroy()
    {
        // This is optional step but if you like you
        // can write hitCount value in your database.
    }
}

```

Now let us compile above servlet and create following entries in web.xml

```

....
<servlet>
    <servlet-name>PageHitCounter</servlet-name>
    <servlet-class>PageHitCounter</servlet-class>
</servlet>

<servlet-mapping>
    <servlet-name>PageHitCounter</servlet-name>
    <url-pattern>/PageHitCounter</url-pattern>
</servlet-mapping>
....

```

Now call this servlet using URL <http://localhost:8080/PageHitCounter>. This would increase counter by one every time this page gets refreshed and it would display following result:

TOTAL NUMBER OF HITS

6

Hit Counter for a Website:

Many times you would be interested in knowing total number of hits on your whole website. This is also very simple in Servlet and we can achieve this using filters.

Following are the steps to be taken to implement a simple website hit counter which is based on Filter Life Cycle:

- Initialize a global variable in `init()` method of a filter.
- Increase global variable every time `doFilter` method is called.
- If required, you can use a database table to store the value of global variable in `destroy()` method of filter. This value can be read inside `init()` method when filter would be initialized next time. This step is optional.

Here I'm assuming that the web container will not be restarted. If it is restarted or servlet destroyed, the hit counter will be reset.

Example:

This example shows how to implement a simple website hit counter:

```
// Import required java libraries
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;

public class SiteHitCounter implements Filter{

    private int hitCount;

    public void init(FilterConfig config)
        throws ServletException{
        // Reset hit counter.
        hitCount = 0;
    }

    public void doFilter(ServletRequest request,
        ServletResponse response,
        FilterChain chain)
        throws java.io.IOException, ServletException {

        // increase counter by one
        hitCount++;

        // Print the counter.
        System.out.println("Site visits count :"+ hitCount );

        // Pass request back down the filter chain
        chain.doFilter(request,response);
    }
    public void destroy()
    {
        // This is optional step but if you like you
        // can write hitCount value in your database.
    }
}
```

Now let us compile above servlet and create following entries in web.xml

```
....
<filter>
    <filter-name>SiteHitCounter</filter-name>
    <filter-class>SiteHitCounter</filter-class>
</filter>

<filter-mapping>
    <filter-name>SiteHitCounter</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>

....
```

Now call any URL like URL <http://localhost:8080/>. This would increase counter by one every time any page gets a hit and it would display following message in the log:

```
Site visits count : 1
Site visits count : 2
Site visits count : 3
Site visits count : 4
Site visits count : 5
.....
```